App Inventor as a Developing Tool to Increase the Accessibility and Readability of Information: A Case Study

Barbara Leporini*

ISTI-CNR, via Moruzzi, Pisa (Italy), barbara.leporini@isti.cnr.it

Giuseppe Catanzaro

Università di Pisa, Largo B. Pontecorvo, Pisa (Italy), g.catanzaro@studenti.unipi.it

Abstract. In this work, App Inventor is presented as a potential tool to develop an accessible app in order to convey contents and information. As use case, we considered the readability of a leaflet used to provide useful information to the general public. Usually this type of contents is presented in a static PDF format, which cannot be easily read on a touch screen. In this work, App Inventor is used to convey information in a more interactive and readable way via a mobile app. The study was specifically aimed at investigating(1) the accessibility support provided by App Inventor, and (2) the usage of an interactive mobile app as a possible tool to enhance content readability on a touch-screen device. The designed app showed that accessibility is supported by App Inventor, although some minor issues have been detected in the user interface design. Finally a set of possible design suggestions has been proposed.

CCS CONCEPTS • Human-centered computing~ Human computer interaction (HCI)~Interactive systems and tools~User interface programming • **Human-centered computing~Accessibility~Accessibility design and evaluation methods**

Additional Keywords and Phrases: Readability, App inventor, Mobile accessibility, screen reader

ACM Reference Format:

Barbara Leporini, Giuseppe Catanzaro. 2020. App Inventor as a Developing Tool to Increase the Accessibility and Readability of Information: A Case Study. In DSAI 2020, December 2–4, 2020, Online, Portugal, 8 pages.

1 INTRODUCTION

Reading information is nowadays a very common task required by everyone. In particular, a large amount of contents is available and accessed via the web. One of the mostly widely-used formats to convey information to the general public is a PDF (Portable Document Format), which is a static presentation with textual contents and images. Unless the contents have been structured in a specific way (e.g. headings and specific tags) [3], the format is generally read in sequential order. Sighted people usually scan the contents and focus on headings or any other graphical elements and styles that catch their attention. Partially sighted people usually encounter some difficulties in replicating this type of activity. Furthermore, blind or severely visually impaired people face a lot of issues in reading a digital document in general [2]. People with cognitive disabilities can also find it difficult to read a PDF due to the static presentation of the information. This occurs especially

DSAI 2020, December 2–4, 2020, Online, Portugal

ACM ISBN 978-1-4503-8937-2/20/12…$15.00
https://doi.org/10.1145/3439231.3440615

when reading a PDF document on a touch screen. PDF files are frequently used by public administrations as a means to provide information in an unmodifiable format to be read either on a computer or mobile device, or printed on paper. Unfortunately, in this way accessing the desired information can require a certain amount of time and effort due to the amount of contents it is necessary to read. This especially happens when using a smartphone with a very small screen. Thus it can be seen that the PDF format still presents several accessibility issues, which are often solved by adapting it into other more accessible formats [15]. More recent mobile solutions are oriented to provide contents in a dynamic and interactive mode. We decided to apply this approach to adapt the static contents of a PDF into a more engaging modality.

In this study, we investigate (1) how the App Inventor framework can be used to adapt a PDF document contents into a suitable and agile application in order to read and access it in a simple and interactive way, including via screen reader. At the same time, (2) the accessibility support provided by the App Inventor framework in delivering the app interfaces for screen reading users has been also investigated in the study.

For our purposes, a PDF leaflet providing health information has been considered as a case study in our work. The leaflet is to be read by travelers before visiting countries which require specific vaccinations, The mobile app developed in this work, named "Traveling Informed", is the result obtained through the design created via App Inventor. The "Traveling Informed" app includes the information usually printed on a PDF leaflet. The reader can obtain information on the necessary vaccinations, based on the geographical area of the destination. The leaflet provides all the information during the vaccination process as well as supporting the traveller in case of need.

2 RELATED WORKS

In literature, a variety of methods and tools have been proposed to make a document or eBook accessible to everyone, including visually-impaired people, like those proposed in [3]. The EPUB format is not yet suitable for being read easily and simply, especially by screen reader users. Various works focus on the accessibility and usability of the reading tasks [10] and [13]. The study in [7] discusses text-customization needed by visually-impaired users in order to read PDF documents. The work in [12] suggests an approach to personalize EPUB visualization so as to adapt the rendering for sighted, visually-impaired and blind people. The results of the studies have encouraged better text customization functionalities in reading tools and eBooks. With the recent release of reading applications, authors are able to develop highly interactive e-books. In fact, by engaging directly with the individual reader, interactive e-books offer the chance to widen knowledge, improve learning as well as enhance the entertainment value of the book [1]. Unfortunately, even recent studies on the personalization of interactive eBooks (in EPUB 3 format) confirm that the problems of interaction via screen reader continue to exist [8].

In our study we plan to investigate another way to design an interactive tool so as to provide contents to all types of users, including screen reading users. In order to do this, a mobile app designed to increase the readability of information via touch-based devices.

Many studies investigated how to develop mobile apps accessible to people with vision impairment [4], [5], [11] and [14], but no work has taken into account the App Inventor as developing tool. In this work we evaluate the accessibility level supported by App Inventor in creating mobile apps. A case study to enhance the reading content has been used to this purpose.

3 THE METHODOLOGY

A PDF leaflet has been adapted into a more flexible mobile app in order to improve the accessibility and readability of its contents on a touch-screen device. The content has been restructured and rearranged so that the user is guided through the information in a more interactive way.

The App inventor framework by MIT (appinventor.mit.edu) has been used as an end-user development tool. App Inventor is a developing framework for those who are not particularly skilled in ICT development and especially in coding. It allows the designer to develop the app through graphical interaction and block coding. All the functions can be handled by the tools made available by the framework. However, developers may not take into consideration accessibility features. Consequently, the framework should be able to apply accessibility features to the user interface (UI) when generating the app. The goal was to investigate the accessibility support offered by this environment especially when people with no specific ICT skills use it to develop an app.

4 THE APP PROTOTYPE

In this section, the app prototype designed to adapt the PDF contents into a more flexible reading tool is described.

4.1 Leaflet Description

For our purpose, a PDF leaflet provided by a local health authority has been used. The leaflet contains information on health data and instructions useful for travelers particularly when travelling to countries where specific vaccinations are either compulsory or optional. In order to avoid getting some vaccinations but missing out on others, the user could be interested in storing information on the vaccinations already administered or to be administered. Reading this leaflet, either in a paper or PDF format, may require a certain effort by the reader; moreover, some information could also be confusing due to the large amount of contents to be read in sequential order, as it may require the reader to retain and refer to previously read information. The brochure is presented as a guide on what to do before a medium / long-term trip: the first part provides some tips for preparing the journey. The second part is dedicated to the vaccinations required and / or suggested for the geographical area chosen as a destination. In addition, more precise details are provided on how the necessary prophylaxis and vaccinations differ for each country. The leaflet also explains the reasons why some countries require mandatory vaccination without which entry into them is prohibited. The third part provides useful tips to avoid infections from certain diseases and indications on how they are generally transmitted.

The leaflet therefore consists of textual information and images. In particular:

Static content: the PDF version reports the content in digital format, but it is not able to offer the opportunity to keep track of the individual's needs, vaccinations administered and any personalization of the trip.

Information overload: the document contains a large amount of information which can make reading inefficient, as it must be done in an almost sequential manner. For example, navigating through the numerous countries in order to acquire the desired information requires time and navigation between the contents in an almost sequential manner.

Lack of customization and configuration for reading. Usually, the PDF does not allow you to change the size, font type or any color contrast. These features are very important for the reader, especially if they are visually impaired or otherwise have difficulty reading. Having an interactive version, on the other hand, can increase the accessibility and readability of the numerous pieces of information.

With the designed app “Traveling Informed”, the content has been arranged and presented to the user in a graphical and interactive way. The app allows the user to interactively navigate through the contents to get the desired information

while exploring the contents, as well as making or selecting some options. In this way, the contents are shown step by step in a contextual modality according to the user's selection and current status.

4.2 App Architecture

The app has been structured in such a way as to facilitate user interaction and content navigation. The app is designed to provide information as the user personalizes their itinerary and makes their choices. In this way, the information is provided on an ad hoc basis, according to the user's preferences, settings and selections. For this reason, the structure consists of two macro areas: "Itinerary" and "Profile", which are available on the home page.

In the "Itinerary" panel the user can create their own itinerary and view the various vaccinations to be carried out for various countries of the world. In the "Profile" panel, the user can enter their personal data, including vaccinations carried out up to the current moment, and add all the information about other people they are travelling with. This means that the app is able to store and show data for more than one user. Through the app, the user is interactively guided through the text by receiving the necessary information from time to time.

4.3 Readability and Accessibility Features in the User Interfaces

The user interface has been designed to be very simple in its contents and components. In a very first version, accessibility was not specifically considered during the block coding in order to evaluate the results obtained from a common coding procedure. After a quick pilot test to evaluate the accessibility while interacting via screen reader, a second version was designed to improve the readability and thus the accessibility. Figure 1 shows an example of a user interface on the app.

Figure 1: "Traveling Informed" app interface

In the app design, some elements and features were considered to support the context and the navigation via screen reader, and to enhance the readability as well. More specifically, some features and elements have been added to the user interface: a navigation bar, information coding and edit boxes for long lists.

4.3.1 Navigation bar.

A navigation bar has been added at the top of each screen in order to aid a screen reader user to better orientate themselves on each page. In fact, a screen reading usually starts from the upper left corner and explores the screen in a sequential way via specific screen reader touch-based gestures [6]. Therefore, a navigation bar consistently available at the beginning of each page can be a very useful orientating element for the screen reader user. Features indicating information

such as current position or main functions located here can improve screen reader interaction. Therefore, the navigation bar contains:

- *Current position*. The navigation bar itself can provide information on the current page (the context), i.e. the current screen. This allows the reader to know their precise whereabouts, so as to better move between the interfaces, as well as the contents. This is particularly useful for screen reading users who can quickly get information on the current page. Information like "'home", "itinerary", etc. can give a context to the current screen. This is useful for everyone, but especially for a screen reading user.
- *Back button*. Specific buttons have been included in the navigation bar located at the top of the screen (page). The "back" button has been specifically located and remains on the left in all the screens in the various interfaces. This is particularly useful for screen reading users who learn that the button is located in that specific position, and can therefore quickly locate it. It is also useful for sighted people for the same reason. This is an important usability property.
- *Contextualized-buttons*. Some buttons located in the same navigation bar at the top change according to the current context. The buttons located on the right side of the navigation bar trigger different actions according to the current page (context). For example, the button "legend", "add a person", "emergency call", etc. are placed on the right of the navigation bar. In this way, the user learns that the current contextualized buttons are located on the right of that navigation bar on each page.

4.3.2 Information coding.

To make the information related to vaccinations easier to read in a more intuitive manner, different solutions have been used in the design:

- *Colors*. Different colors were used to mark and highlight crucial information. Since information about vaccinations is a focus on the content provided by the app (i.e. the leaflet), the "legend" button has been designed to be clearly visible on each screen of the itinerary area; in this way, this type of information can be accessed at any time. The "Legend" area is intended to remind users of the degree of importance (mandatory, requested, recommended and optional) attributed to each vaccination, indicated by a specific color (red, black, gray and green). Figure 2 reports an example: the red color indicates that the vaccinations is compulsory, the others are marked in black which indicates they are requested.
- *Special symbols*. For accessibility reasons, it was necessary to convey this type of information not only through color but also via a special symbol and a short description for screen reading users. The special symbol can be personalized in a settings panel by the user.
- *Hidden labels*. The value attributed to each type of vaccination can be read by the screen reader thanks to a descriptive label placed before the vaccine itself. This label is not visible on the screen but is correctly detected by the reader, which will first read this value and then the vaccine (Fig. 2). Thus, the hidden label is for screen reading users, while the special symbols are for those people who may face issues in seeing or recognizing colors. We decided to use two different modalities because the screen reader can have some problems in correctly detecting some special symbols; this can be due to the screen reader punctuations level set by the user.

4.3.3 Accessing long lists.

In getting information about the itinerary, the user is guided when choosing the country in which he/she is interested. This procedure can be very challenging due to a large number of countries to be explored. All the countries' names could

be reported in a long list. This requires the user to scroll through the list to find the desired one. This task might be boring and time-consuming for every user, but especially for screen reading users. A blind user needs to repeat one gesture for each element available on the list. A command to scroll the list more quickly can also be used, but this might be a problem for non-skilled screen reading users. To this end, we tried to group the countries first by continents, then by countries in alphabetical order. Despite this arrangement, the procedure can nevertheless be tedious. For this reason, a search box has been added in the "Country selection" area, in addition to the list of countries (Fig. 3). Through this search edit box, the user can edit the first letters to show all the countries starting with those letters.

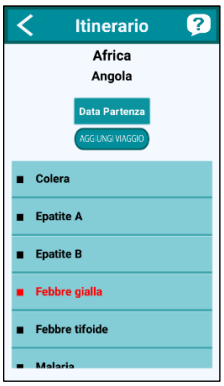

Figure 2: Different colors and legend for coding information

Figure 3: List and search edit interaction

5 ACCESSIBILITY FEATURES

In order to investigate the accessibility level provided by the App Inventor in developing an accessible user interface, some principles proposed for the web sites in [9] and usability requirements for mobile devices [14] have been considered. Specifically, the following aspects have been considered in our mobile app while developing the UI:

Logical order of the UI components. In terms of accessibility via screen reader, a correct order between the items needs to be ensured when interacting via gestures. This has been achieved by positioning each interactive component and textual contents in the proper order, i.e. in a sequence in which the elements should be read by the screen reader. This occurs for both interactive components and sequential order for textual and graphical contents.

One of the examples is given by the positioning of the icons used to remind/indicate to the user whether a vaccination has already been a carried out or not. Visually it would be preferable to show those icons very close to each vaccine name in the list of vaccines in order to have a quick and immediate overview of the vaccinations done. Unfortunately, this can make the reading experience harder via screen reader when accessing the list of vaccines. In fact, the screen reader user will hear something like “vaccine1 name: already done for user 1, already done for user 2, to do for user 3, vaccine2 name: already done for user1, to do for user2, etc.”. The number of users depends on the users recorded in the app. Instead, showing them on the page of specific vaccination, the reading of the list via screen reader is simple, even if the visual reading is slightly penalized. Figure 4 reports the two cases. This feature has been designed in an accessible way with no particular effort by the block coding.

Alternative descriptions for images. For some buttons or information provided in a graphical way, an alternative description was needed. Unfortunately, App Inventor is not able to support the addition of alternative descriptions for images and graphical elements. The solution was to define a specific box in which the image has been added as a

background, and the label has been written in a small font size and with property “textcolor” with value “none” as “hidden labels”. Alternative descriptions are a very important feature in the accessibility principles. Therefore, adding them to the images or graphical elements should be made easier for the designers.

Consistency. Aspects related to the user interface consistency have been specifically considered for the positions of the buttons assigned to each screen.

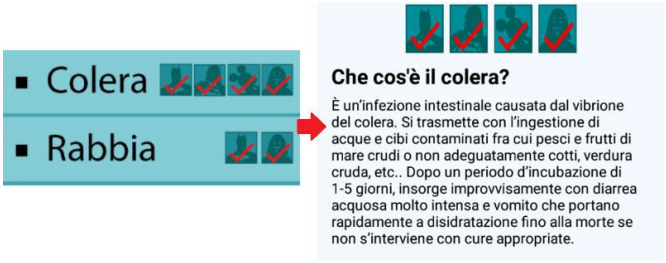

Figure 4: User’s vaccination status: solution more visually suitable solution (on the left), and more accessible via screen reader (on the right)

6 EVALUATION

User testing was performed on a sample of 20 sighted and 8 blind subjects. The users were initially invited to familiarize themselves with the App. Next, each user was asked to perform 6 tasks that summarize the potential experience of the user. The time for task accomplishment was recorded for each user, with the maximum time for each task being three minutes. The difficulties encountered by the users during the use of the App were observed by a researcher, in order to capture any considerations and concerns. Users could ask for clarification while carrying out a task, but could not receive help or suggestions to achieve a specific goal. The six tasks the users had to accomplish were: 1. Create your ID and check cholera and yellow fever vaccinations; 2) Add another traveler and check cholera and yellow fever vaccinations; 3) Look for the “legend” inside the “Itinerary” area; 4) Create a trip to Burkina Faso; 5) Verify if your checked vaccinations have been added to your travel plan; 5) Verify if your trip has been added to “my trip” area.

The tests carried out by the sighted users did not reveal any particular problems. All the users accomplished all the tasks in the allowed time (3 minutes per task). On the contrary, those carried out by the blind users reported some issues. One important issue was related to entering the dates. All the problems highlighted were analyzed and will be resolved in future versions of the application design.

7 DESIGN SUGGESTIONS

Through our experience, some suggestions can be provided to the designers. They can be summarized as follows:

(1) Design a navigation bar at the top of each screen. This is useful as an orientation contextual element, especially for screen reading users.

(2) Make the main and important information always available (or easily reachable) on all the various screens. This helps the reader to quickly acquire the information on the main focused contents related to the core topic provided by the app.

(3) Code crucial information through various techniques. Use, for example, colors, symbols and textual labels (if necessary, hidden) to simplify the immediate comprehension of the crucial and important information by different types of users. Screen reading users can read the textual labels, while colors or symbols are made available for sighted or partially sighted people.

(4) Provide additional ways to access large amounts of information. For example, use alternative or complementary tools to scroll long lists, such as edit boxes for reducing the elements to navigate and read.
(5) Locate and keep specific buttons in fixed positions. This helps the users to locate them and move around more quickly while navigating through the screens and the app.
(6) Structure the contents into more than one page or screen. This is useful to have screens with few contents per page. This makes the readability clearer for everyone, including screen reading users.

8 CONCLUSIONS

This study pointed out that App Inventor framework can be used also by non-skilled ICT developers to produce a very simple and accessible app. Some accessibility issues still exist in the user interface code production related to the alternative descriptions for graphical elements. This issue has been already discussed with the team by MIT in order to improve this.

At the same time, we have also discussed with the MIT team other possible improvements to apply in the user interface designs. This refers to the case in which the developer is a sighted person. Six potential suggestions have therefore been proposed for the designers of user interfaces for apps like the one we considered in this work. Several problems still occur for screen reading developers. So, the opportunity to consider also the accessibility for block coding also via screen reader has been discussed with the team as well.

REFERENCES

[1] Biňas, M., Štancel, P., Novak, M., & Michalko, M. (2012, November). Interactive eBook as a supporting tool for education process. In Emerging eLearning Technologies & Applications (ICETA), 2012 IEEE 10th International Conference on (pp. 39-44). IEEE.

[2] Buzzi, M. C., Buzzi, M., Leporini, B., Mori, G., & Penichet, V. M. (2010, July). Accessing Google docs via screen reader. In International Conference on Computers for Handicapped Persons (pp. 92-99). Springer, Berlin, Heidelberg.

[3] Calabrò, A., Contini, E., & Leporini, B. (2009, November). Book4All: A tool to make an e-book more accessible to students with vision/visual-impairments. In Symposium of the Austrian HCI and Usability Engineering Group (pp. 236-248). Springer, Berlin, Heidelberg.

[4] Carvalho, L. P., & Freire, A. P. (2017, October). Native or web-hybrid apps? An analysis of the adequacy for accessibility of android interface components used with screen readers. In Proceedings of the XVI Brazilian Symposium on Human Factors in Computing Systems (pp. 1-10).

[5] Chiti, S., & Leporini, B. (2012, July). Accessibility of android-based mobile devices: a prototype to investigate interaction with blind users. In International Conference on Computers for Handicapped Persons (pp. 607-614). Springer, Berlin, Heidelberg.

[6] Damaceno, R. J. P., Braga, J. C., & Mena-Chalco, J. P. (2018). Mobile device accessibility for the visually impaired: problems mapping and recommendations. Universal Access in the Information Society, 17(2), 421-435.

[7] Henry, S. L. (2012, July). Developing text customisation functionality requirements of PDF reader and other user agents. In International Conference on Computers for Handicapped Persons (pp. 602-609). Springer, Berlin, Heidelberg.

[8] Leporini, B., & Meattini, C. (2019). Personalization in the Interactive EPUB 3 Reading Experience: Accessibility Issues for Screen Reader Users. In Proceedings of the 16th Web For All 2019 Personalization-Personalizing the Web (pp. 1-10).

[9] Leporini, B., & Paternò, F. (2002, October). Criteria for usability of accessible web sites. In ERCIM Workshop on User Interfaces for All (pp. 43-55). Springer, Berlin, Heidelberg.

[10] Nakai, R., Ito, K., Yanagi, H., & Mima, Y. (2019, July). HapTalker: E-book User Interface for Blind People. In International Conference on Human-Computer Interaction (pp. 282-288). Springer, Cham.

[11] Park, K., Goh, T., & So, H. J. (2014). Toward accessible mobile application design: developing mobile application accessibility guidelines for people with visual impairment. 한국 HCI 학회 학술대회, 31-38.

[12] Schwarz, T., Rajgopal, S., & Stiefelhagen, R. (2018, July). Accessible EPUB: Making EPUB 3 Documents Universal Accessible. In International Conference on Computers Helping People with Special Needs (pp. 85-92). Springer, Cham.

[13] Shin, H., Gil, Y. H., Yu, C., Kim, H. K., Lee, J., & Jee, H. K. (2017). Improved and accessible e-book reader application for visually impaired people. In SIGGRAPH Asia 2017 Posters (pp. 1-2).

[14] Siebra, C., Gouveia, T., Macedo, J., Correia, W., Penha, M., Silva, F., ... & Florentin, F. (2015, November). Usability requirements for mobile accessibility: a study on the vision impairment. In Proceedings of the 14th International Conference on Mobile and Ubiquitous Multimedia (pp. 384-389).

[15] Suzuki, M., & Yamaguchi, K. (2020, September). On Automatic Conversion from E-born PDF into Accessible EPUB3 and Audio-Embedded HTML5. In International Conference on Computers Helping People with Special Needs (pp. 410-416). Springer, Cham.